NASA Technical Paper 3082

May 1991

Guidance, Navigation, and Control Subsystem Equipment Selection Algorithm Using Expert System Methods

Cheryl L. Allen

NASA

NASA
Technical
Paper
3082

1991

Guidance, Navigation, and Control Subsystem Equipment Selection Algorithm Using Expert System Methods

Cheryl L. Allen
Langley Research Center
Hampton, Virginia

NASA

National Aeronautics and
Space Administration
Office of Management
Scientific and Technical
Information Division

Abstract

Enhanced engineering tools can be obtained through the integration of expert system methodologies and existing design software. The application of these methodologies to the Spacecraft Design and Cost Model (SDCM) software provides an improved technique for the selection of hardware for unmanned spacecraft subsystem design. The Knowledge Engineering System (KES) expert system development tool was used to implement a smarter equipment section algorithm than that which is currently achievable through the use of a standard data base system. The Guidance, Navigation, and Control subsystem of the SDCM software was chosen as the initial subsystem for implementation. The portions of the SDCM code which compute the selection criteria and constraints remain intact, and the expert system equipment selection algorithm is embedded within this existing code. This paper will describe the architecture of this new methodology and report on its implementation. The project background and a brief overview of the expert system are described, and once the details of the design are characterized, an example of its implementation is demonstrated.

Acronyms and Abbreviations

BAYES	Bayes' theorem
CDPI	communication, data processing, and instrumentation
DEC	Digital Equipment Corporation
EOS	Earth Observing System
FORTRAN	FORTRAN Version 7.0
GNC	Guidance, Navigation, and Control
HT	hypothesize and test
I/O	input/output
KES	Knowledge Engineering System
LaRC	Langley Research Center
NASA	National Aeronautics and Space Administration
PS	production rules
RIM	Relational Information Management
SDCM	Spacecraft Design and Cost Model
TRW	TRW Space & Technology Group
VMS	Virtual Memory System

Introduction

As the space program enters the 1990's, much attention is being given to the development of unmanned spacecraft which will aid in the study of planet Earth. A resurgence of activity focused on obtaining a better understanding of the Earth's environment has resulted in the proposal and definition of a number of NASA programs. These programs involve various spacecraft with requirements ranging from communication and tracking satellites to large Earth science platforms. The Earth Observing System (EOS) (ref. 1) will employ a large polar-orbiting platform supporting high-precision, Earth-monitoring science instruments. The Mission to Planet Earth program (ref. 2) describes a contingent of spacecraft in both lower Earth orbit and geostationary orbit. These and other similar programs increase the demand placed on the spacecraft design engineer to produce a variety of specialized spacecraft.

In order to increase the efficiency of the design task, the development of advanced computer-aided design and analysis tools has become a necessity. Tools are needed to synthesize spacecraft, test and integrate subsystems, and provide information about on-orbit performance. The Langley Research Center (LaRC) has been heavily involved in the preliminary design and analysis of both manned and unmanned Earth-orbiting spacecraft. One of the many computer programs used to accomplish this task is the Spacecraft Design and Cost Model (SDCM) (ref. 3). This model produces equipment lists of off-the-shelf and projected hardware for the major spacecraft subsystems (including stabilization and control, propulsion, communications, data processing, and thermal control) based upon mission description inputs supplied by the user.

Although SDCM is a versatile tool for performing trade studies, several limitations of the model diminish the reliability of the results. Most notably, the accuracy and completeness of the SDCM design are limited by the accuracy and completeness of the user-supplied input data. Because the model is used to design a complete spacecraft, the user has to have knowledge about each individual subsystem in order to make reasonable assumptions about the mission input. In an attempt to reduce the demand on the subsystem engineer to obtain knowledge outside his specialty, the individual subsystems of the program were separated and modified to run as stand-alone units. While reducing the problems associated with a subsystem expert having to be knowledgeable of a number of different disciplines, program weaknesses still exist. The algorithms responsible for computing

TEST CASE FOR SDCM GPS SPACECRAFT
MISSION DATA

0.20170E+08	1 APOGEE - ORBIT APOGEE(M)	[500E+08]
6.0000	2 C - AXIAL LAUNCH ACELERTIO(G)	[10.0]
0.60000	3 E - LATERAL AUNCH ACCELERATION(G)	[5.0]
2.80	4 DIAMAX - MAXIMUM SAELLITE DIAMETER(M)	[10.]
0.10000E+11	5 OMEGAR - SPIN RATE OF ROTOR	[1.E10]
0.00000E+00	6 OPSMS - NUMBER OF MISION OPS(OPS/S)	[0]
0.96000	7 ORBINC - ORBITAL INCLIATION(RAD)	[0]
0.10000-01	8 PDOTAV - AV. BODY RATE LOW ORBIT(CMG ONLY)(RAD/S)	[.01]
0.10000E-01	9 PDOTRX - REQUIRED SYSTEM RATE ACC. X(RAD/S)	[012]
0.10000E-01	10 PDOTRY - REQUIRED SYSTEM RATE ACC. Y(RAD/S)	[.012]
0.10000E-01	11 PDOTRZ - REQUIRED SYSTEM RATE ACC.Z(RAD/S)	[.012]
0.20170E+08	12 PERIGE - ORBIT PERIGEE(M)	[500E+08]
1800.0	13 SLMX - MAXIMUM SYSTEM WEIGHT(KG)	[1000]
0.23328E+09	14 T - MISSION LIFETIME(SEC)	[60]
1.000	15 TPMIN - MIN P/L SCAN PERIOD(SEC)	[10]
63.000	16 TSMALL - MAIN ENGINE BURN TIME(SEC)	[10]
0.5800E+09	17 TSTAB - PERIOD OF ACTIVE STABILIZATION(SEC)	[0]
0	18 IELORB - 12-HR ELLIPTICAL. ORBIT	[0]
1	19 ISATOR - ORIENTATION 1=EO 2=SO 3=IO	[1]
0	20 MOD - 0=EXPENDABLE SATELLIE 1=MODULARIZED	[0]
0	21 MOEQB - NUMBER OF MODULES IN EQUIPMENT BAY	[0]
1	22 NADIR - NADIR COVERAGE FLAG	[0]
24	23 NFV - NUMBER OF FLIGHT VEHICLES	[4]
1	24 NQV - NUMBER OF QUALIFICATION VEHCLES	[1]
1	25 NSHTL - 1=SATELLITE FLOWN ON SHUTTLE	[0]

Figure 1. Representative SDCM input data base entry.

the selection criteria are sound, but the selection process itself is faulty.

In order to enhance the program's capabilities and provide the design engineer with the state of the art in software tools, a new method that takes advantage of the emerging technologies in the expert system arena is being developed for subsystem equipment selection. After presenting some background into the project and a brief overview of the expert system technology chosen, this paper describes the architecture of this new methodology and reports on its implementation.

Background

Computer-aided design and analysis tools have become an indispensable part of the spacecraft design process. The objective of the SDCM software package is to provide a methodology for developing balanced designs that interrelate cost, performance, safety, and schedule considerations for spacecraft subsystems. The SDCM program uses a two-step process to meet this goal. First, SDCM selects all hardware designs which satisfy the given performance requirements. Once that is accomplished, the model estimates the cost and schedule required to design, build, and operate each spacecraft design. The first step in this process relies on logical and accurate algorithms for equipment selection, and the second step largely depends on the accuracy of the information contained in the data base of equipment descriptions that is associated with the model.

The SDCM software was first developed by the Aerospace Corporation in 1976 (ref. 3) specifically for the design of unmanned, automated spacecraft subsystems. Performance requirements and constraints are calculated based upon mission inputs which are held in a data base and manipulated through the use of an input editor. Figure 1 is representative of the kinds of mission inputs an end user of SDCM would need to provide. The equipment descriptions are contained in a separate data base which lists each hardware option with its technical characteristics and physical properties.

In recent times, the model has experienced a number of transformations. In 1988 the equations and equipment data base of SDCM were expanded by personnel at LaRC and the TRW Space & Technology Group (TRW) to include advanced spacecraft and space station analyses. Subsequently, the program has been divided into five stand-alone modules (one module for each subsystem) thus reducing the complexity of the overall model. Most recently, expert system techniques are being applied to improve the hardware selection process of these individual modules.

The first subsystem implementing this new technique is the Guidance, Navigation, and Control (GNC) subsystem. The GNC subsystem stabilizes the spacecraft to a desired accuracy about a tracking line from a reference point on the vehicle to an external reference. The external reference may be the local vertical of a planet, the Sun, or a more distant star; an inertial reference; or the line of sight to a natural phenomenon like a gravity gradient vector or the lines of the Earth's magnetic field. The accuracy required for attitude stabilization depends upon the purpose of the mission. The performance of the GNC subsystem depends upon the design tradeoffs involving accuracy, average available power, the vehicle's moments of inertia, and the maximum disturbing torques. Hardware is selected based upon its ability to meet the demand of the technical requirements determined by the calculations performed upon the input parameters. Once all the equipment with the qualifying technical characteristics is singled out, the physical attributes of the piece of hardware come into play. For example, if two pieces of equipment can equally meet the technical requirements, then the one which weighs the least may be more desirable. Although this report highlights the GNC implementation, the basic theories and principles can be applied to any one of the disciplines included in SDCM.

Expert System Technology

An expert system is a computer program that uses knowledge and reasoning techniques to solve problems that normally require human evaluation. Like conventional programs, expert systems usually perform relatively well-defined tasks. However, unlike conventional programs, expert systems also explain their actions, justify their conclusions, and provide details of the knowledge they contain.

An important subset of the general area of expert systems concentrates on explicitly representing an expert's knowledge about a class of problems and then providing a separate reasoning mechanism (usually called an inference engine) that operates on this knowledge to produce a solution. These kinds of systems are called knowledge-based expert systems. The knowledge base is a file which contains the facts and heuristics that make up the expert's knowledge. An inference engine is a program that applies knowledge about a specific domain to known facts (as defined by the knowledge base) in order to draw conclusions. Inference engines vary according to the representation of the knowledge and the strategy for applying the knowledge.

At first glance, a knowledge base may appear to be no more than a sophisticated data base; however, further inspection will prove a knowledge base to be far more powerful. Data bases were originally developed to manage records containing large volumes of data. Knowledge about a specific domain may be represented by the structure of the data base through the description of the entities and relations, but the actual contents of the data base are the facts, data, or information, rather than knowledge. Expert systems, on the other hand, are more directly related to solving problems and are not restricted to maintaining records. A knowledge base consists of all the methods an expert uses to perform a task, including computer programs, theories, logic, rules of thumb, and any other number of approaches.

There are a variety of expert system development tools available which assist programmers in building powerful systems capable of solving a wide range of problems. A survey of the market led to the selection of Software Architecture & Engineering's (Software A&E) Knowledge Engineering System (KES) as a development tool (refs. 4–7). The KES tool provides the inference engines, knowledge representation schemes, and facilities for creating an end-user interface. The package also lends itself to integration with existing software.

Because reasoning methods vary with the application, KES provides three inference engines for controlling the use of the knowledge in the knowledge base. The inference engines are production rules (PS), hypothesize and test (HT), and Bayes' theorem (BAYES). The KES PS inference engine uses production rules to represent knowledge and is well suited to applications where domain knowledge is in the form of branching logic or if-then rules. KES PS uses deductive reasoning as the method of problem solving, where certain outcomes follow directly from certain inputs. The outcome of a specific problem can be viewed as a subset from the set of all possible outcomes. PS systems are useful in situations where heuristic "rules of thumb" knowledge is appropriate.

KES HT is a higher level inference engine that is most useful in diagnostic and classification problems. HT simulates reasoning through hypotheses formulation and subsequent verification using abductive reasoning techniques. In abductive reasoning, the conclusion is the most probable explanation of the known premises. The knowledge is represented in framelike descriptions consisting of a collection of statements related to the domain. A principle known as minimal set covering is used to provide as outcome the smallest number of solutions to explain all the known specifications of the problem.

Finally, the KES BAYES inference engine performs statistical pattern classification in support of statistical analysis based on Bayes' theorem. Preexisting knowledge based on the data collected from previous cases is used to determine the likelihood of certain events. BAYES is especially useful in situations where there is a large body of data expressed as probabilities.

In addition to the flexibility KES provides by the choice and/or combination of inference engines, another powerful feature of the system is the ability to integrate the expert system with other software. Depending on the requirements and constraints, either KES can be embedded in another program or KES can communicate with other programs through externals. When using externals, KES communicates with other programs through the management of text files. KES is embedded in other programs by coding function calls within the existing software. With embedding, KES becomes part of a single executable program, allowing information to be passed through memory.

Method

There are five tasks associated with expert system development: analyzing the requirements, acquiring the knowledge, designing the expert system, building the knowledge base, and evaluating the expert system. While analyzing the requirements, the purpose and general goal of the system are defined. The problem to be solved is identified, the context for use of the system is described, and determinations about the input/output (I/O) requirements and end-user interface are made. The second task, acquiring the knowledge, is the most critical phase in the development process because it determines the system's inferential capabilities. The information extracted during this task is used to develop the means by which a problem is solved. During the design phase, the end-user interface is planned, the relationships between the information obtained during the knowledge acquisition phase are determined, and the inferencing technique(s) is chosen. The last two tasks, building the knowledge base and evaluating the expert system, are analogous to the traditional coding and testing phases applied in conventional programming. Although there appears to be a natural sequence for performing these tasks, in reality a significant overlap exists. At any given point in the development process, one or more of these tasks will require more resources than the others.

One last point that needs mentioning prior to the description of the design and development process used to build the GNC equipment selection system is the role that prototyping plays in expert system development. Building a prototype system allows exploration of all the aspects of system development before embarking on a full-scale commitment to any of the earlier tasks. Prototyping highlights potential difficulties and incorrect assumptions before significant resources have been invested in the project.

The tasks outlined above served as a framework for the development of the GNC subsystem equipment selection algorithm. During the requirement analysis task, it was determined that more informed hardware selections could be made than were currently being achieved by SDCM. The scope of the initial project was to be limited to the GNC subsystem, using a specific GNC configuration, the dual-spin satellite configuration, as a prototype. The resources identified for information to define and populate the knowledge base were the existing FORTRAN code and equipment data base, the original SDCM documentation, and in-house subsystem experts. The end users are to be the current SDCM users, and therefore every attempt was to be made to keep the user interface intact and running as the current user community expected. This meant leaving the same input methods and option menus as previously coded.

The hardware platform selected for system development was a Digital Equipment Corporation (DEC) MicroVAX running the VMS operating system; however, the KES development tool (and therefore the expert system) supports a large number of host machines. The SDCM program resides on a DEC VAX 11/785 which is networked to the MicroVAX through a common file server system.

Aside from the inadequacies already delineated in the equipment selection algorithms, the calculations performed in SDCM to determine spacecraft characteristics and requirements are well tested and therefore trusted. The KES knowledge base would have to be developed in such a way as not to interfere with this part of the program. These calculations play an integral part in the preparation for equipment selection and therefore serve as a major knowledge source during the knowledge acquisition task. Other valuable sources for the domain knowledge came from the SDCM manuals, resident GNC subsystem human experts, and the relationships already defined in the equipment data base.

The equipment data base contains hardware listed by part number for each subsystem considered in SDCM. Attributes and relations are defined, and

EQUIPMENT IDENTIFICATION NUMBER:	1206
SUBSYSTEM:	STABILITY AND
CONTROL	
EQUIPMENT TYPE:	EARTH SENSOR
TECHNICAL CHARACTERISTICS:	
(1) SENSOR NOISE	0.1047E-02
(2) RADIANCE IRREGULARITY (DEG)	0.4362E-03
(3) QUANTIZATION ERROR	0.1047E-02
(4) SUN INTERFERENCE (DEG)	0.8725E-03
(5) MOON INTERFERENCE (DEG)	0.0000E+00
(6) THRESHOLD AGING (DEG)	0.0000E+00
(7) NULL OR BIAS ERROR (DEG)	0.0000E+00
(8) MAXIMUM OUTPUT FREQUENCY (RAD/SEC)	0.0000E+00
(9)	
(10)	
POWER:	
AVERAGE POWER(WATTS)	0.6500E+01
MAXIMUM POWER(WATTS)	0.7500E+01
MINIMUM POWER(WATTS)	0.0000E+00
NOMINAL VOLTAGE(VOLTS)	0.1500E+02
MAXIMUM VOLTAGE(VOLTS)	0.3000E+02
MINIMUM VOLTAGE(VOLTS)	0.5000E+01
CONVERTER/INVERTER REQUIREMENT(FLAG)	0.1413E+04
WEIGHT(KG):	0.4432E+01
VOLUME(M**3)	0.9820E-02
VIBRATION:	
RANDOM(C)	0.2047E+02
NON-RANDOM(C)	0.0000E+00
TEMPERATURE:	
MAXIMUM(DEG-K)	0.3270E+03
MINIMUM(DEG-K)	0.2548E+03
PRESSURE(PA):	0.0000E+00
CDPI:	
POWER SWITCHING COMMANDS(NO)	0.1000E+01
TIME TAGGED COMMANDS(NO)	0.0000E+00
OTHER COMMANDS(NO)	0.1000E+01
HIGH RATE TELEMETRY:	
ANALOG POINTS(NO)	0.6000E+01
DIGITAL POINTS(NO)	0.0000E+00
SAMPLE RATE(SEC-1)	0.1000E+02
WORD LENGTH(BITS)	0.8000E+01
LOW RATE TELEMETRY:	
ANALOG POINTS(NO)	0.2000E+01
DIGITAL POINTS(NO)	0.0000E+00
SAMPLE RATE(SEC-1)	0.1000E+00
WORD LENGTH(BITS)	0.8000E+01
RELIABILITY:	
FAILURE MODEL(FLAG)	0.1000E+01
FAILURE RATE(*10**9HR)	0.4250E+04
STANDARD DEVIATION(*10**9HR)	0.0000E+00
DORMANCY FACTOR(NO)	0.5000E+00
TOTAL REDUNDANT ELEMENTS	0.4000E+01
COST:	
DESIGN ENGINEERING($1000)	0.1954E+03
TEST AND EVALUATION($1000)	0.1480E+03
UNIT PRODUCTION($1000)	0.1490E+03
REFERENCE QUANTITY	0.1000E+01
FACTOR(NO)	0.1000E+01
ORIGINAL SPACECRAFT	
MANUFACTURER AND TYPE	

Figure 2. RIM equipment data base entry.

information is stored using the Relational Information Management (RIM) (ref. 9) system data base manager. A typical entry (seen in fig. 2) contains up to 10 technical characteristic entries, as well as values describing the physical properties of a particular piece of equipment. As mentioned previously, equipment selection is based on the ability of a piece of hardware to meet the technical requirements of the spacecraft (currently SDCM selects the first piece of equipment in the data base which meets the computed requirements, not necessarily the best piece of equipment). In order to make a selection, the technical characteristics of the hardware components must be replicated within the KES knowledge base. The original equipment data base will also remain intact and, when integrated with the new system, will serve

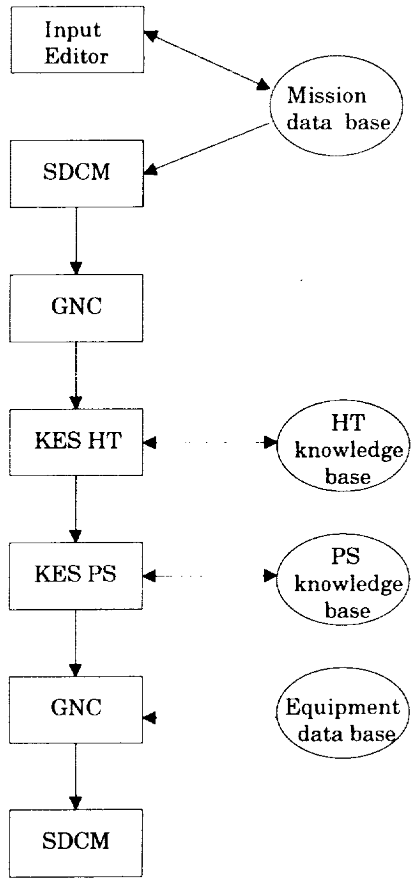

Figure 3. KES/GNC flow diagram.

as the source for the physical properties associated with an equipment selection.

Finally, during the requirements analysis phase, subsystem experts were identified who could furnish the heuristic or intuitive knowledge necessary to make decisions about components that at first glance appeared equally matched. The experts were also responsible for providing confidence in the requirements computations and supplying the knowledge lacking in the current selection algorithms.

The knowledge acquisition task followed directly from the requirements analysis task. Data were gathered from the experts, the existing data base, and the manuals. This task continued throughout the entire development and continues today as program refinements are underway.

The program flow resulting from the design phase for the GNC subsystem is shown in figure 3. In order to fulfill the specifications outlined in the requirements analysis task, the majority of SDCM remained as coded. The input editor and accompanying mission data base were left unchanged so as not to alter the end-user interface. Likewise, the portions of code (both in the SDCM and GNC subsystem modules) which calculate spacecraft characteristics and performance requirements were left unchanged. The part of SDCM replaced by the expert system was the section that made the actual equipment selections. Separate expert system modules were created and called from (embedded) within the GNC subsystem. Once these modules have made their selections, control is once again returned to the GNC module so that subsystem totals for weight, power, and volume can be computed for the selected equipment. The original equipment data base is consulted for these values at the component level. Finally, the SDCM module is activated to format the output as the end user expects.

A close coupling of the FORTRAN code and the KES knowledge base is to be achieved through embedding. By embedding KES within the existing model, a direct link is established through function calls which are placed within the FORTRAN code.

Embedding is a two-step process. First the knowledge base must be built to run as a stand-alone expert system. Once this is done, the stand-alone system and the existing code can be integrated. Function calls are placed within the FORTRAN code and allow the program to instruct KES to send, receive, and manipulate data. KES is also able to ask for input from, and to send messages to, the FORTRAN code. These function calls are provided by KES and are maintained in a library linked to the system during compilation.

A combination of inferencing techniques was chosen to perform the equipment selection. The ability of KES HT to manage classification problems made it a good tool for conducting preliminary assessments about the hardware available for selection. The minimal set covering technique used by HT designates the smallest number of components which meet the technical requirements determined in SDCM. The technical characteristics of each piece of candidate equipment are represented in the knowledge base in framelike descriptions. Figure 4 shows these descriptions for a section of sensors whose technical characteristics include the sensor type, number of axes about which the sensor takes readings, and the sensor accuracy in lower Earth orbit and geostationary orbit (sensor_type, num_of_axes, sal, sag, respectively). Another advantage of using HT is that decisions about equipment can be made with incomplete information. For example, if you have information about the requirement for sensor accuracy in geostationary orbit but are not concerned with this value

```
sensor:mlt
  (Part_9101
   [description:
    sensor_type = asun;
    num_of_axes = one;
    sal = very_coarse;
    sag = very_coarse;],

   Part_9104
   [description:
    sensor_type = dsun;
    num_of_axes = two;
    sal = fine;
    sag = fine;],

   Part_9111
   [description:
    sensor_type = earth;
    num_of_axes = two;
    sal = coarse;
    sag = medium;],
```

Figure 4. Sensor descriptions.

```
if leo_err = 0.0 then
  sal = na.
  else if leo_err le 0.00029 then
  sal = very_fine.
  else if leo_err ge 0.1 then
  sal = very_coarse.
  else if leo_err ge 0.01 then
  sal = coarse.
  else if leo_err ge 0.001 then
  sal = medium.
  else if leo_err ge 0.0001 then
  sal = fine.
  else sal = very_fine.
  endif.
  endif.
  endif.
  endif.
  endif.
endif.
```

Figure 5. Range definition.

at lower Earth orbit, KES HT will choose equipment with technical characteristics which meet the most known facts.

The largest drawback in using the HT inference engine in this application is its inability to handle numerics in the equipment descriptions. By setting up character strings to represent ranges of values as shown in figure 5, all equipment within an acceptable range will be selected.

Once a group of equipment is selected, each piece of which will meet the technical requirements, decisions must be made as to which piece of equipment is optimal for any particular mission. Often, if two components are equally capable, the one weighing the least is chosen. Other parameters, such as minimal power consumption or cost, may also be considered. The KES PS inference engine will be used to aid in these types of decisions.

The KES PS inference engine uses production rules to represent knowledge. It is well suited to applications where if-then logic dominates. The general form of a PS production rule is

```
if
  antecedent
then
  consequent
end if.
```

The antecedent of a rule must be true in order for the consequent to be performed. PS provides the class mechanism to allow elements with similar attributes (same attributes but most likely different values) to be grouped together. Figure 6 shows the class definitions for the sensors and actuators used in the GNC subsystem.

Once SDCM receives the list of potential equipment from the KES HT knowledge base, it will be passed along to the KES PS knowledge base for optimization. The technical requirements are checked numerically to make sure the selected equipment meets or exceeds the desired value, and then the piece of equipment weighing the least is selected. Minimal weight was chosen by the experts as the discriminating parameter; however, the system could be easily modified such that any number of parameters could be used in determining the most appropriate piece of equipment.

After the components are selected, the equipment identification numbers will be passed back through function calls, and SDCM will assume control once more. At this point, the number of necessary components is determined, and values for weight, volume, and power consumption are retrieved from the equipment data base and totaled. The program output presented to the user remains unchanged from the original SDCM in the spirit of maintaining the familiar end-user interface.

Implementation

Many parts of the design have been implemented and tested. A prototype of the HT knowledge base for the GNC subsystem of the dual-spin satellite configuration was built and tested to run in the stand-alone mode. Because the dual-spin configuration is fairly simplistic and presents no particular challenges

```
classes:

  Actuator:
    attributes:
      act_type: sgl (mt,rwa,cmg).
      moment: real.
      mmdb: real.
      gimnum: int.
    %
  endclass.

  Sensor:
    attributes:
      sensor_type: sgl
      (earth,asun,dsun,mmter,star,gyro).
      num_of_axes: int.
      sal: real.
      sag: real.
    %
  endclass.

%
```

Figure 6. Class definitions.

to the system, a decision was made to develop the three-axis stabilized configuration also. The prototype was completed and tested to satisfaction in the stand-alone mode.

Figure 7 shows the output from a test case run on the HT portion of the system using the dual-spin case. The dual-spin spacecraft selects despin electronics, despin mechanisms, control electronics, two sensors (one Earth sensor and one Sun sensor), gimbal electronics, valve drivers, biaxial assemblies, and a nutation damper. The mission input necessary to select the GNC components for this test case describes an Earth-pointing, dual-spin spacecraft in lower Earth orbit. By definition, a dual-spin spacecraft uses four attitude control thrusters. The user sets values for allowable sensor errors based upon the mission objectives. In this test case, sensor errors in lower Earth orbit of up to 0.01° are acceptable. Allowable errors in geostationary orbit are of no consequence for this test case and are therefore set to zero. The projected spin inertia for this spacecraft is computed by SDCM to be 4000 kg-m^2. For this configuration, figure 7 shows single components chosen for the despin electronics, despin mechanism, and control electronics. The symbol <a> after a part number means that, given the current input, this is always the best choice. As can be seen by the list of possible values, only a single choice for each of these equipment types exists. When selecting Sun sensors, KES HT recommends part 9101 as the best possibility but suggests that parts 9102 and 9103 also have a high probability (<h>) of meeting the requirements. More than one component is recommended

```
Name: despin_elec
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_101
Current value:
Part_101 <a>
Inferred: yes
 Inferred from a description

Name: despin_mech
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_103
Current value:
Part_103 <a>
Inferred: yes
 Inferred from a description

Name: control_elec
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_603
Current value:
Part_603 <a>
Inferred: yes
 Inferred from a description

Name: sensor
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_9101
Part_9102
Part_9103
Part_9104
Part_9105
Part_9106
Part_9107
Part_9108
Part_9109
Part_9110
Part_9111
Part_9112
Part_9113
Part_9114
Part_9115
Part_9116
Part_9117
Part_9118
Part_9119
Current value:
Part_9103 <h>
Part_9102 <h>
Part_9101 <a>
Inferred: yes
 Inferred from a description

Name: sensor
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_9101
Part_9102
Part_9103
Part_9104
Part_9105
Part_9106
Part_9107
Part_9108
Part_9109
Part_9110
Part_9111
Part_9112
Part_9113
Part_9114
Part_9115
Part_9116
Part_9117
Part_9118
Part_9119
Current value:
Part_9111
Inferred: yes
 Inferred from a description

Name: gimbal_elec
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_510
Part_503
Current value:
Part_503 <a>
Inferred: yes
 Inferred from a description

Name: valve_driver
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_203
Part_206
Part_209
Part_1601
Part_1602
Part_1605
Current value:
Part_203 <m>
Part_206 <m>
Part_209 <m>
Inferred: yes
 Inferred from a description

Name: biaxial_assem
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_701
Part_703
Part_706
Current value:
Part_701 <a>
Inferred: yes
 Inferred from a description

Name: nutation_damper
Kind of entity: Attribute
Type: mlt
Marked: evoking
Possible values:
Part_403
Part_406
Part_409
Part_412
Part_415
Current value:
Part_409 <a>
Inferred: yes
 Inferred from a description

Enter command: stop
```

Figure 7. Dual-spin test case output.

here because there is no single piece of equipment in the knowledge base that fully meets all the requirements. Referring back to figure 4, sensor descriptions contain values for four attributes: sensor type, number of axes, and allowable errors in lower Earth (sal) and geostationary (sag) orbit. Part 9101 meets the type and axis specification, but an allowable error of 0.01° would fall into the coarse range, not the very coarse range characteristic of this part. Parts 9102 and 9103 meet the error requirement but are two-axis systems. Because KES is unable to specify what attribute in the equipment description is the more determinant requirement, all are presented for further evaluation. The second sensor selected is the Earth sensor. Part 9111 satisfies more of the requirements than any other component and therefore is the only one presented to the user. For valve drivers, parts 203, 206, and 209 will each do equally well. Single selections are made for the biaxial assembly and the nutation damper because, in each case, one of the components meets all of the requirements.

Progress is being made on the PS knowledge base both in the areas of definition and implementation. The next step will be to begin the integration of the two independent systems with the SDCM code. The KES-supplied function calls will have to be modified to serve the needs of this particular application. This work is underway as this report goes to print.

Finally, the elegance of the subsystem design is a reflection of the equipment data base from which the design algorithm has to choose. An update of the equipment data base (and henceforth the knowledge bases) is necessary to lend more credibility to the program's results. Much of the equipment is outdated, going back to SDCM's conception in the early 1970's. TRW added space station components during the task assignment of 1988, but much of this information is incomplete or representative of technology forecasts rather than off-the-shelf equipment. A separate task to improve the equipment data base is essential to the success of the selection process as it currently stands.

Concluding Remarks

Expert system technologies are being applied to existing design software in an attempt to enhance the tools currently available to the design engineer. This report demonstrates an application of these new techniques for improving the equipment selection capabilities of the Spacecraft Design and Cost Model (SDCM). The equipment selection algorithm in SDCM is faulty, and the introduction of a more logical approach gained through the application of an expert system increases the reliability of the software system by eliminating existing limitations in the selection process.

The definition and design of the new system are complete, and implementation is well underway. By maintaining portions of the existing FORTRAN code and embedding the newly developed stand-alone expert systems, the integration task can be performed in a relatively short period of time. Using both abductive (hypothesize and test) and deductive (production rules) inferencing methodologies, both classification and optimization can be achieved. Building prototypes of the expert systems allows new ideas to be tested in advance, increasing the confidence in the design, the new techniques, and eventually the completed system.

The equipment selected during the spacecraft design task should represent the state-of-the-art, off-the-shelf hardware. In order to make this happen, the current equipment data base needs to be revised and then regularly maintained.

NASA Langley Research Center
Hampton, VA 23665-5225
March 8, 1991

References

1. *EOS—A Mission to Planet Earth.* NASA Headquarters, 1990.

2. Ride, Sally K.: *Leadership and America's Future in Space.* A Report to the Administrator of NASA, Aug. 1987.

3. Advanced Mission Analysis Directorate, Advanced Orbital Systems Division: *Systems Cost/Performance Analysis (Study 2.3) Final Report. Volume II: Systems Cost/Performance Model.* Aerosp. Rep. No. ATR-75 (7363)-3, Vol. II (Contract NASW-2727), Aerospace Corp., Mar. 31, 1975. (Supersedes and replaces ATR-74(7343)-1, Vol. II, dated Sept. 27, 1974, and includes Addendum, dated June 1976.)

4. *Knowledge Base Author's Manual—KES PS.* Software Architecture & Engineering, Inc., c.1990.

5. *Knowledge Base Author's Reference Manual—KES PS.* Software Architecture & Engineering, Inc., c.1990.

6. *Knowledge Base Author's Manual—KES HT.* Software Architecture & Engineering, Inc., c.1988.

7. *Knowledge Base Author's Reference Manual—KES HT.* Software Architecture & Engineering, Inc., c.1988.

8. Boeing Computer Services: *RIM—Version 7, Relational Information Management System and Report Writer User's Guide.* Central Scientific Computing Complex Doc. Z-4, May 1986.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No. NASA TP-3082	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Guidance, Navigation, and Control Subsystem Equipment Selection Algorithm Using Expert System Methods		5. Report Date May 1991
		6. Performing Organization Code
7. Author(s) Cheryl L. Allen		8. Performing Organization Report No. L-16896
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665-5225		10. Work Unit No. 506-49-21-02
		11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001		13. Type of Report and Period Covered Technical Paper
		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

Enhanced engineering tools can be obtained through the integration of expert system methodologies and existing design software. The application of these methodologies to the Spacecraft Design and Cost Model (SDCM) software provides an improved technique for the selection of hardware for unmanned spacecraft subsystem design. The Knowledge Engineering System (KES) expert system development tool was used to implement a smarter equipment selection algorithm than that which is currently achievable through the use of a standard data base system. The Guidance, Navigation, and Control subsystem of the SDCM software was chosen as the initial subsystem for implementation. The portions of the SDCM code which compute the selection criteria and constraints remain intact, and the expert system equipment selection algorithm is embedded within this existing code. This paper will describe the architecture of this new methodology and report on its implementation. The project background and a brief overview of the expert system are described, and once the details of the design are characterized, an example of its implementation is demonstrated.

17. Key Words (Suggested by Author(s)) Expert system Equipment selection	18. Distribution Statement Unclassified—Unlimited Subject Category 61

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 10	22. Price A03

NASA FORM 1626 OCT 86

NASA-Langley, 1991

For sale by the National Technical Information Service, Springfield, Virginia 22161-2171